Meta-GAIN for Missing Data Imputation

Tao Tong[1], **Xiaofeng Zhu**[1, 2], **Jiangzhang Gan**[2*]

[1] School of Computer Science and Technology
University of Electronic Science and Technology of China, Chengdu 611731, China
[2] School of Computer Science and Technology, Hainan University, Haikou 570228, China
tongqtao@gmail.com, seanzhuxf@gmail.com, ganjzgxnu@163.com

Abstract

Although previous deep imputation methods (*e.g.,* Generative Adversarial Network (GAN) based methods) have been widely designed to impute missing data, they still suffer from the issues, *i.e.,* lack of the imputation diversity and the generalization ability. In this paper, we propose a new GAN-based imputation method, namely Meta-based Generative Adversarial Imputation Network (Meta-GAIN), to investigate a new generator for achieving diverse imputation and generalization ability. Specifically, we employ the Kullback-Leibler (KL) divergence to achieve the imputation diversity by generating a continuous embedding space of the original data. We also design a task regularizer to suppress redundant features and capture a more authentic distribution, thus enhancing the generalization ability of the imputation model. Moreover, we theoretically prove that our proposed regularizer achieves the generalization ability. In addition, we design a new meta network to efficient optimize our objective function as well as to improve imputation diversity. Experimental results on real datasets show that our method outperforms all comparison methods under different missing mechanisms in terms of imputation and classification performance.

Introduction

Missing values are often found in real-world applications such as medical data analysis and financial data application, due to various reasons like privacy, lost, and so on (Adhikari et al. 2022; Bryzgalova et al. 2025). Deep learning on incomplete datasets (*i.e.,* where a part of feature values are missing) makes existing deep learning methods limited and inapplicable as these methods are usually designed for dealing with complete datasets whose feature values are observed. As a result, an increasing attention is focused on conducting deep learning with missing values. Missing data imputation which involves estimating values for the missing data to fill incomplete datasets is becoming one of the most popular methods among all solutions for dealing with missing values. The reason is that the imputed values make existing deep methods available (Sun et al. 2023; Seu, Kang, and Lee 2022; Alwateer et al. 2024).

Missing data imputation methods can be broadly categorized into three subgroups, *i.e.,* statistical imputation methods, traditional imputation methods and deep imputation methods. Due to their powerful feature extraction ability, deep imputation methods are more popular than both statistical methods and traditional methods in real applications. To be more specific, deep imputation methods leverage deep neural networks to automatically learn hierarchical representations and generate sophisticated estimations for missing data especially for high-dimensional data. For example, Generative Adversarial Imputation Nets (GAIN) is designed to use the Generative Adversarial Network (GAN) to impute missing data with the help of the hint matrix (Yoon, Jordon, and Schaar 2018). Graph Imputation Neural Network (GINN) first treats the data as a graph, and then uses a neural network to infer and fill in the missing values based on the connections and patterns observed in the graph structure (Hammad Alharbi and Kimura 2020). Due to the remarkable proficiency in estimating data distributions, deep imputation methods especially GAN-based methods are very popular in real applications. Further details on missing data imputation methods are provided in supplementary Material.

Although previous deep imputation methods have achieved significant imputation performance, there are still some limitations to be handled. First, deep imputation methods (*e.g.,* GAN-based methods) mainly focus on imputation accuracy by ignoring the imputation diversity. However, diversity is crucial for missing data imputation because it ensures the imputed data more authentically reflect the original data distribution, avoiding bias or limiting the accuracy of subsequent model training. As a result, these GAN-based methods without imputation diversity may lead to poor flexibility. For example, the methods (*i.e.,* GAIN (Yoon, Jordon, and Schaar 2018) and meshGAN (Hammad Alharbi and Kimura 2020)) perform well in the training set, but their imputation performance decreases with the increase of the missing ratio, resulting in limited adaptability. Second, many deep imputation methods primarily leverage observed values, but they are irrelevant to downstream tasks and thus influencing the generalization ability of the model. For example, the methods (*i.e.,* DAGAN (Liu et al. 2021) and MIA (Josse et al. 2024)) utilize labels as supervision to guide missing data imputation. However, they are not designed to consider the generalization ability of the model.

To address the above issues, in this paper, we propose a new GAN-based imputation method, namely Meta-GAIN,

*Corresponding author
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

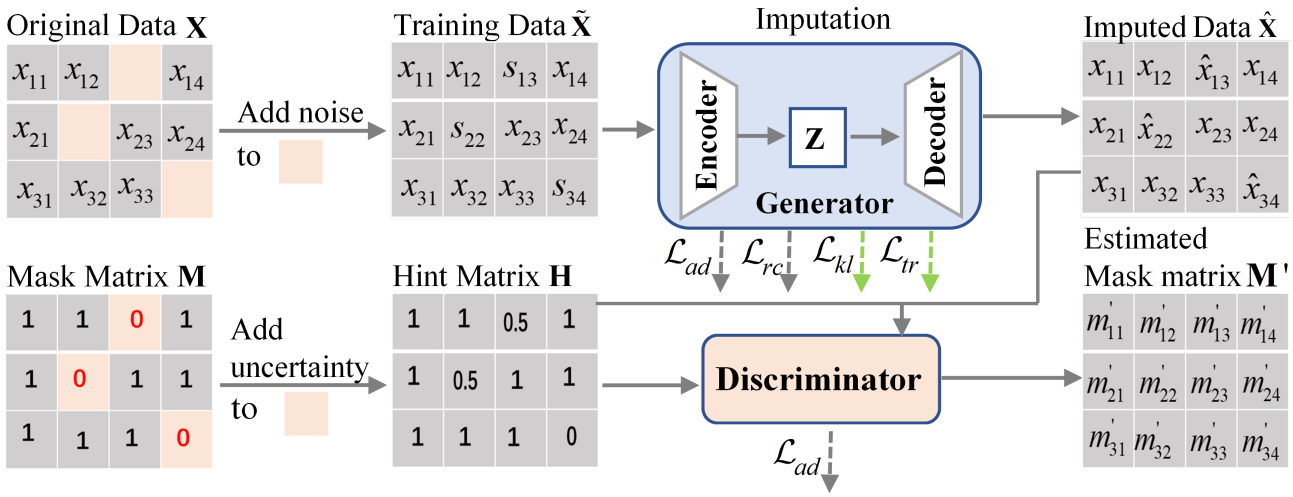

Figure 1: The framework of the proposed Meta-GAIN. Given the original data $\mathbf{X}$, we first obtain its mask matrix $\mathbf{M}$, and then add random noise and uncertainty, respectively, to obtain $\tilde{\mathbf{X}}$ and $\mathbf{H}$. After the generator involving the reconstruction loss (*i.e.,* $\mathcal{L}_{rc}$), the regularizer for the imputation diversity (*i.e.,* $\mathcal{L}_{kl}$), the task regularizer (*i.e.,* $\mathcal{L}_{tr}$), and the adversarial loss (*i.e.,* $\mathcal{L}_{ad}$), we obtain the embedding $\mathbf{Z}$, the decoder which is used to impute missing values, and the imputed data $\hat{\mathbf{X}}$. Finally, we input both $\hat{\mathbf{X}}$ and $\mathbf{H}$ to the discriminator with the discriminator loss $\mathcal{L}_{ad}$ to obtain the estimated mask matrix $\mathbf{M}'$.

shown in Figure 1. Similar to GAN-based imputation methods, our method includes a generator and a discriminator, but we mainly focus on designing a new generator to achieve imputation diversity and generalization ability. Specifically, we employ the Kullback-Leibler (KL) divergence (Joyce 2011) to make the embedding space of the original data continuous, which enhances the diversity of imputed data (McCoy, Kroon, and Auret 2018), thus exploring the first issue. We also design a task regularizer (*i.e.,* the cross entropy loss between the classification results obtained by the embedding and the ground truth) to reduce redundant features to approximate the authentic distribution, and thus mitigating the overfitting issue and enhancing the generalization ability (Poulos and Valle 2018). Moreover, we theoretically prove that the proposed task regularizer achieves the generalization ability, thus exploring the second issue of previous methods. In addition, we design a new meta network to adaptively optimize the hyper-parameters of our proposed objective function, which enables our model to be adaptable and improves the imputation diversity.

Compared to previous imputation methods, the main contributions of our method can be summarized as follows:

- We propose a new method to achieve imputation diversity. Specifically, the KL divergence makes the embedding space of the original dataset continuous to achieve imputation diversity. Besides, the meta network also improves the imputation diversity by adaptively learning hyper-parameters.
- We propose a new task regularizer to generate task related representations, thus improving the generalization ability. We also theoretically analyze that how the proposed task regularizer achieves the generalization ability.
- Experimental results demonstrate the effectiveness of the proposed method under different missing mechanisms on different datasets with different noisy ratios.

Method

Denoting $\mathcal{D}_{data} = \mathcal{D} \cup \mathcal{D}_{me}$ as the dataset with n samples whose feature number is d, $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$ and $\mathcal{D}_{me} = (\mathbf{X}_{me} \in \mathcal{R}^{n\times d}, \mathbf{Y}_{me})$, respectively, are the training data and the meta data. Besides, $\mathbf{x}_i$ is the i-th training sample and $y_i \in \{0,1\}^C$ is the corresponding label with C classes. Given the dataset $\mathcal{D}_{data}$, GAN-based imputation methods aim at training an imputation network (*i.e.,* the decoder) based on the generator G and the discriminator $\mathbb{D}$.

Motivation

Missing data imputation attempts to replace missing values with accurate estimation. For example, GAIN (Yoon, Jordon, and Schaar 2018) revises GAN to conduct missing data imputation by the following objective functions:

$$\min_{G} \max_{\mathbb{D}} \; V(\mathbb{D}, G), \tag{1}$$

where

$$\begin{aligned} V(\mathbb{D}, G) =& \mathbb{E}_{\hat{\mathbf{X}},\mathbf{M},\mathbf{H}}\big[\mathbf{M}^T \log \mathbb{D}(\hat{\mathbf{X}}, \mathbf{H}) \\ &+ (1-\mathbf{M})^T \log(1 - \mathbb{D}(\hat{\mathbf{X}}, \mathbf{H}))\big], \end{aligned} \tag{2}$$

where $\hat{\mathbf{X}} \in \mathcal{R}^{n\times d}$ is the imputed dataset of the original data $\mathbf{X}$. $\mathbf{M} \in \{0,1\}^{n\times d}$ is the mask matrix (as known as the indicator matrix) of $\mathbf{X}$, where 1 means observed value and 0 means missing value. $\mathbf{H} \in \mathcal{R}^{n\times d}$ is the hint matrix, whose missing values are randomly imputed. The loss functions of

G and D in (Yoon, Jordon, and Schaar 2018) are listed as follows:

$$\mathcal{L}_{Ge}(\mathbf{X}, \hat{\mathbf{X}}, \mathbf{M}, \hat{\mathbf{M}}) = \underbrace{\sum (\hat{\mathbf{x}}_i - \mathbf{x}_i)^2}_{\mathcal{L}_{rc}} - \underbrace{\eta \sum (1 - \mathbf{m}_i) \log(\hat{\mathbf{m}}_i)}_{\mathcal{L}_{ad}}, \quad (3)$$

$$\mathcal{L}_D(\mathbf{M}, \hat{\mathbf{M}}, \mathbf{H}) = \sum [\mathbf{m}_i \log(\hat{\mathbf{m}}_i) + (1 - \mathbf{m}_i) \log(1 - \hat{\mathbf{m}}_i)], \quad (4)$$

where $\hat{\mathbf{M}} \in \mathcal{R}^{n \times d}$ is the estimated mask matrix, $\mathcal{L}_{rc}$ is the reconstruction loss, and $\mathcal{L}_{ad}$ is the adversarial loss of GAIN. GAIN employs a generator (*i.e.,* an Multilayer Perceptron (MLP) to impute missing values and a discriminator to distinguish real data from imputed data.

However, as aforementioned, GAIN uses an MLP as the generator to suffer from the issues, including imputation diversity and generalization ability. In this paper, we propose the Meta-GAIN in Figure 1 to address the above issues.

Meta-GAIN

The proposed Meta-GAIN is designed to train an imputation network (*i.e.,* the encoder network ϕ_{en}), based on the generator G (including an encoder network, a task regularizer ϕ_{tr}, and a meta network $\boldsymbol{\theta}$), and the discriminator D with the discriminator network ϕ_D.

Imputation Diversity The goal of imputation diversity is to prevent the model from overfitting to the current data and to enhance the robustness of the imputed data for downstream tasks. Previous methods tend to maintain the imputation diversity by either adding random noise to the training data or designing a particular regularizer. For example, (Karras et al. 2020) add random noise to the features of each layer in the generator, which is equal to introduce additional random dimensions into the embedding space, thus expanding the effective sampling scope to enhance imputation diversity. (Tseng et al. 2021) apply an $\mathcal{L}_2$ regularizer to weaken the discriminator capability, enriching the gradient directions to achieve the imputation diversity. Although these methods can achieve imputation diversity, they still have disadvantages. For example, adding noise may cause the imputed data to deviate from the original data. The $\mathcal{L}_2$ regularizer may compresses the expressive capacity of the embedding. To address this issue, we investigate a KL divergence to make the distribution of the embedding space continuous, which is available to enhance the imputation diversity of the model.

Give the training data $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$ with missing values, we first divide the original dataset into multiple batches, and then randomly add noise into the training data to get the new matrix $\tilde{\mathbf{X}}$ of $\mathbf{X}$, *i.e.,*

$$\tilde{\mathbf{X}} = \mathbf{M} \odot \mathbf{X} + (1 - \mathbf{M}) \odot \mathbf{S}, \quad (5)$$

where $\odot$ is the element-wise multiplication operator and $\mathbf{S} \in [0, 1]^{n \times d}$ is the random noise matrix to avoid the generator consistently output either mean value or fixed value. To achieve the imputation diversity, we consider to control the distribution of the embedding $\mathbf{Z}$ of $\tilde{\mathbf{X}}$ in the encoder by:

$$\mathcal{L}_{kl} = \mathrm{KL}(p(\mathbf{Z}|\tilde{\mathbf{X}}) \parallel p(\mathbf{Z})), \quad (6)$$

where $p(\mathbf{Z})$ is the true prior distribution of $\mathbf{Z}$ and $p(\mathbf{Z}|\tilde{\mathbf{X}})$ is the true posterior distribution of the embedding $\mathbf{Z}$ on $\tilde{\mathbf{X}}$.

The KL divergence in Eq. (6) reduces the distribution difference between $p(\mathbf{Z})$ and $p(\mathbf{Z}|\tilde{\mathbf{X}})$ to ensure that the learned embedding space distribution approaches a continuous prior distribution, thereby making the embedding space continuous. As a result, this makes the imputation process (*i.e.,* imputed values are sampled from the embedding space) flexible, and thus improving the imputation diversity. Moreover, it explores the issues in previous methods. For example, the proposed KL regularizer restricts the deviation between the latent distribution and the prior distribution, thus reducing the risk of imputation values deviating from the original data. Meanwhile, by constraining the posterior distribution from $\tilde{\mathbf{X}}$ to $\mathbf{Z}$, the embedding space retains its continuous characteristics and probabilistic expressive capacity, thus exploring the issue of the methods with the $\mathcal{L}_2$ regularizer.

However, $p(\mathbf{Z}|\tilde{\mathbf{X}})$ in Eq. (6) is incomputable because its computational cost is exponential explosion with the increase of the dimensionality. Hence, in this paper, we follow (Odaibo 2019) to use the conditional probability distribution (*i.e.,* $q_{\phi_{en}}(\mathbf{Z}|\tilde{\mathbf{X}})$) of the embedding $\mathbf{Z}$ given the original data $\tilde{\mathbf{X}}$ to approximately estimate $p(\mathbf{Z}|\tilde{\mathbf{X}})$, so the regularizer for the imputation diversity in Eq. (6) is changed to:

$$\mathcal{L}_{kl} = \mathrm{KL}(q_{\phi_{en}}(\mathbf{Z}|\tilde{\mathbf{X}}) \parallel p(\mathbf{Z})), \quad (7)$$

Because $q_{\phi_{en}}(\mathbf{Z}|\tilde{\mathbf{X}})$ is a manually designed, parameterizable distribution, *i.e.,* a standard Gaussian distribution, it has a closed-form solution and its parameters are directly output by the encoder network. Therefore, it is easy to compute.

Task Regularizer Although our method solves the issues in previous methods (*e.g.,* GAIN) to achieve imputation diverisity for missing data imputation, but previous methods often ignore the generalization ability of the imputation, resulting in bad imputation performance on out-of-distribution data. For example, the unsupervised adversarial loss in GAIN only constrains the fidelity of reconstruction and lacks task-relevant regularization. This causes the generator to easily fit the unique noise in the training data rather than the underlying data distribution, thereby leading to overfitting and poor generalization ability of the imputation model. To solve this issue, we propose a task regularizer on the embedding $\mathbf{Z}$ to achieve the generalization ability.

Specifically, in the generator, given $\mathbf{Z}$ as the input, we employ an MLP as the network of the downstream task (*i.e.,* classification tasks) and regard the cross entropy loss as the task regularizer, *i.e.,*

$$\mathcal{L}_{tr} = \mathbb{E}_{(\tilde{\mathbf{X}}, \mathbf{Y})} \left[-\log f_{\phi_{tr}}(\mathbf{Y}|\mathbf{Z}) \right], \quad (8)$$

where $f_{\phi_{tr}}$ is the downstream task (*e.g.,* classification) with parameters ϕ_{tr}.

After introducing the label information by Eq. (8), our method constrains the imputation results to the task space through the label consistency, which is equivalent to imposing data-independent prior noise. This forces the model to learn the general structure of samples rather than the unique noise specific to the training data, thereby suppressing the

overfitting issue to improve the generalization ability of the model. Detailed information regarding the theoretical analysis of the proposed task regularizer can be found in Subsection "Theoretical Analysis".

Objective function After considering the imputation diversity by $\mathcal{L}_{kl}$ and the generalization ability by $\mathcal{L}_{tr}$, we combine them with the adversarial loss $\mathcal{L}_{ad}$ and the reconstruction loss $\mathcal{L}_{rc}$ to have the final objective function $\mathcal{L}_G$ of our method as follows.

$$\mathcal{L}_G = \mathcal{L}_{rc} + \beta\mathcal{L}_{kl} + \gamma\mathcal{L}_{tr} + \eta\mathcal{L}_{ad}, \tag{9}$$

where β, γ, and η and hyper-parameters.

Optimization by Meta Network It is usually difficult to tune the hyper-parameters in Eq. (9). Previous works typically set them with fixed values, easily resulting in the lack of flexibility. For example, HIVAE (Nazabal et al. 2020) manually sets its hyper-parameters and $\beta - VAE$ (Higgins et al. 2017) sets its hyper-parameters with the methods of linear annealing. In this paper, we design a meta network to adaptively learn the hyper-parameters in Eq. (9), *i.e.,* β, γ, and η. Specifically, based on Model-agnostic Meta Learning (MAML) (Finn, Abbeel, and Levine 2017), our meta network includes the inner, *i.e.,* the base learner $\mathcal{L}_G(\mathbf{W})$ where $\mathbf{W}$ is denoted as all network parameters in the generator, and the outer, *i.e.,* the meta learner $\mathcal{L}_{me}$ with the parameter $\boldsymbol{\theta}$. Furthermore, we define the objective function $\mathcal{L}_{me}$ of our meta network as follows:

$$\mathcal{L}_{me} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i^{me} - \hat{\mathbf{x}}_i^{me})^2 \tag{10}$$

where $\mathbf{X}^{me}$ is the training samples of the meta network and $\hat{\mathbf{X}}^{me}$ is the imputed data of the original meta samples.

In this paper, we design a bi-level minimization method to optimize our objective function in Eq. (9):

$$\begin{cases} \mathbf{w}^*(\boldsymbol{\theta}) = \arg\min_{\mathbf{w}} \mathcal{L}_G(\mathbf{w}; \boldsymbol{\theta}) \\ \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{me}(\mathbf{w}^*(\boldsymbol{\theta})) \end{cases}. \tag{11}$$

In this paper, we employ Stochastic Gradient Descent (SGD) (Amari 1993) to alternately optimize Eq. (11). As a result, the optimization process of the generator is separated into three steps. Specifically, we first draw a batch of samples from the meta dataset $\mathbf{X}^{me}$, and then update the parameter $\mathbf{w}$ by the following rule:

$$\hat{\mathbf{w}}^{(t)} = \mathbf{w}^{(t)} - \eta_1 \nabla_{\mathbf{w}} \mathcal{L}_G(\mathbf{w}; \boldsymbol{\theta}) \tag{12}$$

After updating the parameters in the generator, *i.e.,* from $\mathbf{w}^{(t)}$ to $\hat{\mathbf{w}}^{(t)}$ by Eq. (12), we update the parameters of the meta network by:

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \eta_2 \nabla_{\boldsymbol{\theta}} \mathcal{L}_{me}(\hat{\mathbf{w}}(\boldsymbol{\theta})) \tag{13}$$

Since the update of $\mathbf{w}$ is guided by the meta network $\boldsymbol{\theta}$, it may lead to inaccurate hyper-parameters with the help of $\boldsymbol{\theta}^t$ instead of $\boldsymbol{\theta}^{t+1}$. Hence, we re-update $\hat{\mathbf{w}}^{(t)}$ to $\mathbf{w}^{(t+1)}$ by Eq. (12). Specifically, during the first calculation of Eq. (12), only the network parameters of the generator are updated, and gradients are not backpropagated. In the second calculation of Eq. (12), all the network parameters are updated while gradients are back-propagated.

By adaptively adjusting hyper-parameters β, γ, and η. by our meta network, our imputation method can maintain sufficient uncertainty in the embedding space while meeting task accuracy constraints, thus enhancing the imputation diversity. The pseudo code of the proposed Meta-GAIN is shown in Algorithm 1.

Theoretical Analysis

In Eq. (8), we design a task regularizer to achieve the generalization ability for missing data imputation. In this section, we theoretically analyze that the proposed task regularizer reduces the generalization error of the downstream task, *e.g.,* the classification task in this paper.

Let $p(\mathbf{X})$ be the true data distribution and $\hat{p}(\mathbf{X})$ be the empirical distribution over n samples, the generalization error for classifier $f_{\phi_{tr}}$ is defined as follows:

$$\begin{aligned} \epsilon_{\text{gen}} = &\big|\mathbb{E}_{(\mathbf{X},\mathbf{Y})\sim p(\mathbf{X})}\left[\ell(\mathbf{Y}, f_{\phi_{tr}}(\mathbf{Z}))\right] \\ &- \frac{1}{n}\sum_{i=1}^{n}\ell(y_i, f_{\phi_{tr}}(\mathbf{z}_i))\big|, \end{aligned} \tag{14}$$

where ℓ is the 0-1 loss. In Eq. (14), the first term is the true risk and the second term is the empirical risk.

Based on (Yin, Kannan, and Bartlett 2019), we apply the Rademacher complexity bound for the function class $f_{\phi_{tr}}$ to have:

$$\epsilon_{\text{gen}} \leq 2\mathfrak{R}_n(\mathcal{F}_{\phi_{tr}}) + \sqrt{\frac{\log(1/\delta)}{2n}} + \mathbb{E}[\ell_{01}] \tag{15}$$

Applying Pinsker's inequality (Caprio 2022) to the 0-1 loss, we have:

$$\mathbb{E}[\ell_{01}] \leq \frac{1}{2}\sqrt{\mathbb{E}\left[D_{\text{KL}}\left(H(\mathbf{Y},\mathbf{Z}) \| H_{\phi_{tr}}(\mathbf{Y},\mathbf{Z})\right)\right]} \tag{16}$$

where $H(\mathbf{Y}|\mathbf{Z})$ is conditional entropy and $H_{\phi_{tr}}(Y|Z)$ is the cross entropy. Their definitions are shown as below:

$$H(\mathbf{Y}|\mathbf{Z}) = -E_{P(\mathbf{Y},\mathbf{Z})}[log\, P(\mathbf{Y}|\mathbf{Z})] \tag{17}$$

$$H_{\phi_{tr}}(\mathbf{Y}|\mathbf{Z}) = -E_{P(\mathbf{Y},\mathbf{Z})}[log\, Q_{\phi_{tr}}(\mathbf{Y}|\mathbf{Z})] \tag{18}$$

where $P(\mathbf{Y},\mathbf{Z})$ is the probability distribution of the true label $\mathbf{Y}$ given the embedding $\mathbf{Z}$. $Q_{\phi_{tr}}(\mathbf{Y}|\mathbf{Z})$ is the conditional probability distribution predicted by the classification network which is actually $\mathcal{L}_{tr}$. Based on Eq. (17) and Eq. (18), Eq. (16) can be rewritten as:

$$\mathbb{E}[\ell_{01}] \leq \frac{1}{2}\sqrt{\mathcal{L}_{\text{tr}} - H(\mathbf{Y}|\mathbf{Z})} \tag{19}$$

Combining Eq. (15) with Eq. (19), we have:

$$\begin{aligned} \epsilon_{\text{gen}} \leq &2\mathfrak{R}_n(\mathcal{F}_{\phi_{tr}}) + \sqrt{\frac{\log(1/\delta)}{2n}} \\ &+ \frac{1}{2}\sqrt{\mathcal{L}_{\text{tr}} - H(\mathbf{Y}|\mathbf{Z})} \end{aligned} \tag{20}$$

Since the value of $H(\mathbf{Y}|\mathbf{Z})$ is independent to the classification network, we can decline the upper bound of the generalization error by optimizing $\mathcal{L}_{\phi_{tr}}$. That is to say, the generalization ability of the imputation model was enhanced with the help of task regularizer.

Algorithm 1: The pseudo-code of the proposed Meta-GAIN.

Input: Training data $\mathcal{D} = \{\mathbf{X}, \mathbf{Y}\}$, meta data $\mathcal{D}_{me} = \{\mathbf{X_{me}}, \mathbf{Y_{me}}\}$, the batch size k_{bs}, the maximal epoch (Max_E).

Output: the decoder network ϕ_{de}.

Initialize the parameters of all networks, *i.e.,* $\mathbf{w}$, $\boldsymbol{\theta}$, and ϕ_{ad} , and the parameters, *i.e.,* β, γ and η.
while $I < Max\_E$ **do**
  Choose k_{bs} samples from $\mathcal{D}$
  Generate k_{bs} random numbers that follow the Bernoulli distribution denoted as B.
  while $j < k_{bs}$ **do**
    $\bar{\mathbf{x}}(j) \leftarrow G(w)$
    $\hat{\mathbf{x}}(j) \leftarrow \mathbf{m}(j) \odot \mathbf{x}(j) + (1 - m(j)) \odot \bar{\mathbf{x}}(j)$
    $\mathbf{h}(j) = \mathbf{b}(j) \odot \mathbf{m}(j) + 0.5(1 - \mathbf{b}(j))$
  end while
  Update discriminator using SGD with $\nabla D - \sum_{j=1}^{k_D} \mathcal{L}_D(\mathbf{m}(j), D(\hat{\mathbf{x}}(j), \mathbf{h}(j)), \mathbf{b}(j))$
  Choose k_{bs} samples from $\mathcal{D}_{me}$
  Generate k_{bs} random numbers that follow the Bernoulli distribution denoted as B_{me}.
  while $j < k_{bs}$ **do**
    $h(j) = \mathbf{b}(j) \odot m(j) + 0.5(1 - \mathbf{b}(j))$
  end while
  Choose k_{bs} samples from $\mathcal{D}_{me}$
  Update generator G with parameter $\mathbf{w}$ by Eq. (12) (from $\mathbf{w}^t$ to $\hat{\mathbf{w}}^t$)
  Update $\boldsymbol{\theta}$ with Eq. (13)
  Update $\mathbf{w}$ with Eq. (12) (from $\hat{\mathbf{w}}^t$ to $\mathbf{w}^{t+1}$)
end while

Experiments

We compare our proposed Meta-GAIN with 9 comparison methods on 11 datasets at different missing ratios, in terms of imputation performance and classification accuracy.

Experimental Setup

Datasets: We conduct experiments on eleven real datasets from UCI dataset[1], whose detailed information is summarized in supplementary material. In our experiments, we follow (Yoon, Jordon, and Schaar 2018) to corrupt the original data with different missing mechanisms, *i.e.,* Missing Completely At Random (MCAR), Missing At Random (MAR), and Missing Not At Random (MNAR) missing mechanism.

Comparison Methods: The omparison methods include two statistical imputation methods (*i.e.,* mean and KNNI (Regression 1992)), three traditional imputation methods (*i.e.,* XGBI (Chen and Guestrin 2016), MissFI (Stekhoven and Bühlmann 2012), and PCAI (Josse, Pagès, and Husson 2011)) and four deep imputation methods (*i.e.,* MIDAE (Gondara and Wang 2018), VAEI (McCoy, Kroon, and Auret 2018), HIVAE (Nazabal et al. 2020) and GAIN (Yoon, Jordon, and Schaar 2018)). The detailed experimental results of extra comparison methods *i.e.,* REMASKER (Du, Melis, and Wang 2023) and HyperImpute (Jarrett et al. 2022) are listed in the supplementary material.

Implementation Details: All experiments are conducted with an Intel i9-12900K CPU and a NVIDIA RTX3090 GPU. We implement the proposed Meta-GAIN with PyTorch framework and the source codes of all comparison methods are collected from the authors, we also set the parameters of all comparison methods by following the corresponding original papers so that they output their best results. In the proposed method, we apply ReLU as the activation function and SGD as the optimizer with a momentum of 0.9. We also set the batch size as 64, the learning weight as 10^{-2}, and gradually decay as 10^{-5}. For the selection fo the meta data, we randomly and equally select 10 % of samples from each class as the meta data. In our network architecture, the networks (including the encoder, the decoder, and the classification network) are a 5-layer MLP and every layer is fully-connected with 100 neurons. All methods employ XGBoost (Chen and Guestrin 2016) as the classifier to evaluate the classification performance on different datasets with the same setting on the same dataset.

Metrics: We employ classification accuracy (Accuracy) to evaluate classification performance, and employ Average Root Mean Square Error (ARMSE) combining the root mean square error with accuracy error to evaluate the imputation performance. More detailed information are listed in supplementary material.

Result Analysis

We list the imputation results and the classification results, respectively, of all methods in Tables 1-3 and Table 4.

First, the proposed method outperforms all comparison methods on all datasets, followed by GAIN, HIVAE, VAEI, MIDAE, MissFI, XGBI, KNNI, and Mean. For example, our method declines on average by 0.031 and 0.128, respectively, compared to the best comparison method (*i.e.,* GAIN) and the worst comparison method (*i.e.,* VAEI) in terms of the evaluation metric ARMSE. This indicates that it is necessary to consider them in a unified framework, *i.e.,* the imputation diversity by the KL divergence, the generalization ability by the task regularizer, and the optimization by the meta network, for missing data imputation.

Second, the proposed Meta-GAIN outperforms all the methods without considering the imputation diversity, (*i.e.,* MIDAE and GAIN). For example, the proposed Meta-GAIN achieves an average increase of 0.028 and 0.024, respectively, compared to MIDAE and GAIN on all datasets in terms of classification accuracy. This verifies that it is reasonable to use the KL divergence to achieve the imputation diversity.

Third, the proposed Meta-GAIN outperforms the deep imputation methods without considering the generalization ability (*i.e.,* MIDAE, VAEI, HIVAEI, and GAIN) under different missing mechanisms. For example, the proposed Meta-GAIN obtains the performance with an average decline of 0.03 and 0.128, respectively, compared to the best (*i.e.,* GAIN) and the worst (*i.e.,* VAEI) on all datasets in terms of the evaluation metric ARMSE. This indicates that

[1]https://archive.ics.uci.edu

Datasets	Mean	KNNI	XGBI	MissFI	PCAI	MIDAE	VAEI	HIVAE	GAIN	Meta-GAIN
Wireless	0.224	0.182	0.184	0.165	0.218	0.195	0.299	0.166	0.131	**0.115**
Yeast	0.222	0.183	0.154	0.206	0.182	0.133	0.130	0.117	0.110	**0.094**
Balance	0.438	0.451	0.453	0.458	0.442	0.466	0.522	0.382	0.414	**0.346**
Valley	0.529	0.511	0.452	0.449	0.451	0.439	0.471	0.383	0.417	**0.395**
Wine	0.228	0.204	0.259	0.207	0.231	0.428	0.218	0.205	0.171	**0.156**
Connect	0.298	0.287	0.286	0.253	0.273	0.268	0.269	0.275	0.273	**0.241**
Letter	0.221	0.132	0.140	0.145	0.187	0.156	0.167	0.160	0.159	**0.113**
Turkiye	0.442	0.230	0.232	0.218	0.244	0.214	0.224	0.356	0.192	**0.175**
Chess	0.392	0.442	0.398	0.432	0.389	0.441	0.356	0.345	0.383	**0.326**
Anuram	0.222	0.123	0.136	0.111	0.144	0.150	0.143	0.162	0.098	**0.081**
Heart	0.347	0.360	0.345	0.325	0.342	0.407	0.288	0.278	0.271	**0.255**

Table 1: ARMSE results of all methods on different datasets at 20% missing ratio under the MCAR missing mechanism.

Datasets	Mean	KNNI	XGBI	MissFI	PCAI	MIDAE	VAEI	HIVAE	GAIN	Meta-GAIN
Wireless	0.261	0.178	0.142	0.152	0.188	0.207	0.321	0.171	0.135	**0.128**
Yeast	0.229	0.203	0.177	0.207	0.212	0.155	0.170	0.135	0.122	**0.121**
Balance	0.367	0.445	0.441	0.489	0.433	0.429	0.523	0.366	0.425	**0.356**
Valley	0.387	0.467	0.478	0.522	0.442	0.438	0.539	0.368	0.402	**0.364**
Wine	0.269	0.207	0.171	0.204	0.240	0.439	0.225	0.208	0.176	**0.165**
Connect	0.311	0.291	0.299	0.267	0.302	0.298	0.341	0.312	0.267	**0.265**
Letter	0.221	0.131	0.188	0.126	0.192	0.172	0.190	0.161	0.144	**0.126**
Turkiye	0.412	0.264	0.254	0.223	0.263	0.225	0.273	0.352	0.311	**0.184**
Chess	0.362	0.417	0.413	0.439	0.418	0.442	0.358	0.346	0.372	**0.338**
Anuram	0.242	0.102	0.172	0.141	0.172	0.190	0.199	0.201	0.199	**0.151**
Heart	0.332	0.342	0.334	0.336	0.332	0.432	0.302	0.289	0.290	**0.279**

Table 2: ARMSE results of all methods on different datasets at 20% missing ratio under the MAR missing mechanism.

the task regularizer can directly improve the generalization ability of the imputation model.

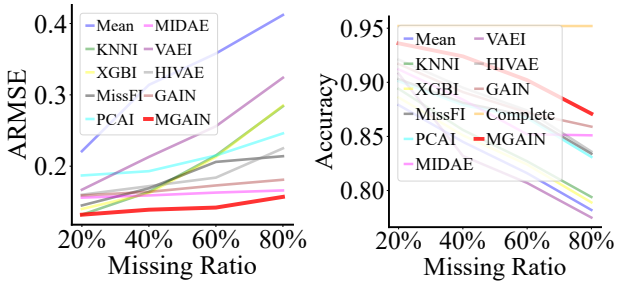

Figure 2: Imputation results and classification accuracy of all methods at different missing ratios on datasets *i.e.,* Letter(the result of datasets Wireless and Anuran are in supplementary materials), under MCAR missing mechanism (MGAIN is the proposed Meta-GAIN).

Last but not least, we investigate the performance of all methods on three datasets (*i.e.,* Letter, Wireless, and Anuran) at different missing ratios by reporting the results in Figure 2 (result on dataset Anuran is in supplementary Material). On one hand, our method still achieves the best, compared to all comparison methods. This indicates the robustness of our method. On the other hand, our method outperforms the deep imputation methods without hyper-parameter adaptation (*i.e.,* MIDAE, VAEI, HIVAEI, and GAIN). For example, the proposed method achieves an average decline of 0.038 and 0.095, respectively, compared to the best comparison method (*i.e.,* GAIN) and the worst comparison method (*i.e.,* VAEI) on all datasets in terms of the evaluation metric ARMSE. Moreover, the proposed method achieves an average increase of 0.016 and 0.044, respectively, compared to the best comparison method (*i.e.,* HIVAE) and the worst comparison method (*i.e.,* VAEI) on all datasets in terms of classification accuracy. This indicates that adaptively adjusting hyper-parameters can achieve adaptability with the help of meta network.

Ablation Study

The proposed method includes three key components, *i.e.,* the imputation diversity (ID), the task regularizer (TR), and the meta network MN)[2]. We investigate the effectiveness of every component by listing the imputation results of 8 methods in Table 5.

First, the method without considering any component (*i.e.,* the second row in Table 5) is worse than any method that only considers one component (*i.e.,* IM, TR, and MN) or even any method in Table 5. This verifies that every component of our method is useful for missing data imputation.

[2]In Table 5, the values of the hyper-parameters (*i.e.,* β, γ, and η) were set as $\beta, \gamma, \eta \in \{0.01, 0.1, 1, 10\}$ if the methods do not have the MN component, *i.e.,* the third row, the fifth row, and the seventh row in Table 5.

Datasets	Mean	KNNI	XGBI	MissFI	PCAI	MIDAE	VAEI	HIVAE	GAIN	Meta-GAIN
Wireless	0.376	0.280	0.295	0.266	0.321	0.360	0.553	0.303	0.261	**0.256**
Yeast	0.312	0.307	0.336	0.282	0.332	0.287	0.294	0.235	0.242	**0.229**
Balance	0.710	0.705	0.702	0.709	0.712	0.776	0.841	0.705	0.611	**0.558**
Valley	0.732	0.752	0.738	0.739	0.734	0.779	0.833	0.711	0.642	**0.584**
Wine	0.430	0.362	0.382	0.368	0.420	0.659	0.462	0.372	0.307	**0.302**
Connect	0.425	0.423	0.372	0.404	0.432	0.542	0.551	0.503	0.403	**0.396**
Letter	0.334	0.298	0.247	0.267	0.303	0.304	0.431	0.252	0.146	**0.141**
Turkiye	0.592	0.498	0.512	0.505	0.342	0.416	0.692	0.528	0.388	**0.352**
Chess	0.641	0.635	0.599	0.621	0.612	0.756	0.637	0.603	0.509	**0.472**
Anuram	0.372	0.256	0.332	0.276	0.298	0.337	0.459	0.342	0.392	**0.275**
Heart	0.587	0.578	0.602	0.583	0.576	0.689	0.633	0.493	0.356	**0.346**

Table 3: ARMSE results of all methods on different datasets at 20% missing ratio under the MNAR missing mechanism.

Datasets	Complete	Mean	KNNI	XGBI	MissFI	PCAI	MIDAE	VAEI	HIVAEI	GAIN	Meta-GAIN
Wireless	0.986	0.923	0.934	0.931	0.943	0.937	0.952	0.949	0.964	0.957	**0.972**
Yeast	0.582	0.491	0.513	0.504	0.521	0.514	0.533	0.526	0.541	0.532	**0.552**
Balance	0.925	0.852	0.861	0.858	0.873	0.867	0.882	0.879	0.891	0.885	**0.902**
Valley	0.535	0.461	0.479	0.473	0.487	0.484	0.494	0.491	0.501	0.498	**0.514**
Wine	0.983	0.912	0.924	0.918	0.934	0.931	0.943	0.939	0.951	0.947	**0.962**
Connect	0.857	0.776	0.787	0.782	0.795	0.792	0.802	0.801	0.809	0.805	**0.914**
Letter	0.952	0.879	0.894	0.886	0.903	0.901	0.912	0.908	0.921	0.917	**0.936**
Turkiye	0.864	0.793	0.798	0.795	0.806	0.803	0.813	0.810	0.821	0.816	**0.835**
Chess	0.903	0.826	0.845	0.834	0.854	0.851	0.862	0.858	0.869	0.865	**0.874**
Anuram	0.981	0.901	0.932	0.912	0.942	0.939	0.951	0.947	0.962	0.956	**0.974**
Heart	0.850	0.788	0.792	0.790	0.801	0.797	0.812	0.806	0.819	0.815	**0.826**

Table 4: Classification accuracy of all methods on different datasets at 20% missing ratio under the MCAR missing mechanism.

ID	MN	TR	Wireless	Yeast	Balance	Valley	Win	Connect	Letter	Turkiye	Chess	Anuram	Heart
×	×	×	0.131	0.110	0.414	0.417	0.171	0.273	0.159	0.192	0.383	0.098	0.271
✓	×	×	0.129	0.100	0.349	0.412	0.168	0.216	0.139	0.189	0.371	0.091	0.267
×	✓	×	0.123	0.097	0.372	0.411	0.164	0.259	0.131	0.185	0.353	0.087	0.263
×	×	✓	0.129	0.100	0.412	0.415	0.170	0.267	0.142	0.191	0.375	0.095	0.270
✓	✓	×	0.118	0.095	0.357	0.398	0.158	0.244	0.119	0.179	0.332	0.083	0.257
✓	×	✓	0.122	0.096	0.366	0.407	0.161	0.251	0.129	0.184	0.369	0.089	0.264
×	✓	✓	0.120	0.096	0.359	0.402	0.160	0.249	0.125	0.182	0.335	0.089	0.261
✓	✓	✓	**0.115**	**0.094**	**0.346**	**0.395**	**0.156**	**0.241**	**0.113**	**0.175**	**0.326**	**0.081**	**0.255**

Table 5: ARMSE results of the ablation study on the dataset Letter at 20% missing ratio under the MCAR missing mechanism.

Second, the methods considering two components (*i.e.,* from the sixth row to the eighth row) are better than any method considering only one component (*i.e.,* IM, TR, and MN). In particularly, our method considering all components outperforms all other methods in Table 5. This demonstrates that it is reasonable to consider all of components, *i.e.,* the imputation diversity, the generalization ability, and the meta network, in a unified framework for missing data imputation.

Third, MN beats other two methods (*i.e.,* ID and TR). For example, MN is with an average decline of 0.007 and 0.011, respectively, on all datasets, compared to ID and TR. This implies that the meta network is the best components out of three ones of our methods. The reason may be that the hyper-parameter adaptation introduces different constraints at different stages to enlarge the embedding space, and thus improving the imputation diversity. Moreover, in the experiments, we always find that the values β is larger than the values of either γ or η. This verifies again for the importance of the meta network in our method.

Conclusion

In this paper, we proposed an new GAN based imputation method to address the issues in previous methods. Specifically, we investigated both the KL divergence and the meta network to improve the imputation diversity, as well as designed the task regularizer to achieve the generalization ability of the imputation model. Moreover, we theoretically analyzed the effectiveness of the proposed task regularizer. Experimental results on real datasets verified the effectiveness of the proposed method in terms of different imputation mechanism at different missing ratios.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFA1004100.

References

Adhikari, D.; Jiang, W.; Zhan, J.; He, Z.; Rawat, D. B.; Aickelin, U.; and Khorshidi, H. A. 2022. A comprehensive survey on imputation of missing data in internet of things. *ACM Computing Surveys*, 55(7): 1–38.

Alwateer, M.; Atlam, E.-S.; Abd El-Raouf, M. M.; Ghoneim, O. A.; and Gad, I. 2024. Missing data imputation: A comprehensive review. *Journal of Computer and Communications*, 12(11): 53–75.

Amari, S.-i. 1993. Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5(4-5): 185–196.

Bryzgalova, S.; Lerner, S.; Lettau, M.; and Pelger, M. 2025. Missing financial data. *The Review of Financial Studies*, 38(3): 803–882.

Caprio, M. 2022. Refined Pinsker's and reverse Pinsker's inequalities for probability distributions of different dimensions. *IEEE Access*, 10: 116425–116431.

Chen, T.; and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Du, T.; Melis, L.; and Wang, T. 2023. Remasker: Imputing tabular data with masked autoencoding. In *International Conference on Learning Representations*, 1–23.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135.

Gondara, L.; and Wang, K. 2018. Mida: Multiple imputation using denoising autoencoders. In *Pacific-Asia conference on knowledge discovery and data mining*, 260–272.

Hammad Alharbi, H.; and Kimura, M. 2020. Missing data imputation using data generated by gan. In *Proceedings of the 2020 3rd International Conference on Computing and Big Data*, 73–77.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*.

Jarrett, D.; Cebere, B. C.; Liu, T.; Curth, A.; and van der Schaar, M. 2022. Hyperimpute: Generalized iterative imputation with automatic model selection. In *International Conference on Machine Learning*, 9916–9937.

Josse, J.; Chen, J. M.; Prost, N.; Varoquaux, G.; and Scornet, E. 2024. On the consistency of supervised learning with missing values. *Statistical Papers*, 65(9): 5447–5479.

Josse, J.; Pagès, J.; and Husson, F. 2011. Multiple imputation in principal component analysis. *Advances in data analysis and classification*, 5(3): 231–246.

Joyce, J. M. 2011. Kullback-leibler divergence. In *International encyclopedia of statistical science*, 720–722. Springer.

Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8110–8119.

Liu, T.; Fan, J.; Luo, Y.; Tang, N.; Li, G.; and Du, X. 2021. Adaptive data augmentation for supervised learning over missing data. *Proceedings of the VLDB Endowment*, 14(7): 1202–1214.

McCoy, J. T.; Kroon, S.; and Auret, L. 2018. Variational autoencoders for missing data imputation with application to a simulated milling circuit. *IFAC-PapersOnLine*, 51(21): 141–146.

Nazabal, A.; Olmos, P. M.; Ghahramani, Z.; and Valera, I. 2020. Handling incomplete heterogeneous data using vaes. *Pattern Recognition*, 107: 1–19.

Odaibo, S. 2019. Tutorial: Deriving the standard variational autoencoder (vae) loss function. *arXiv preprint arXiv:1907.08956*.

Poulos, J.; and Valle, R. 2018. Missing data imputation for supervised learning. *Applied Artificial Intelligence*, 32(2): 186–196.

Regression, N. 1992. An Introduction to Kernel and Nearest-Neighbor. *The American Statistician*, 46(3): 175–185.

Seu, K.; Kang, M.-S.; and Lee, H. 2022. An intelligent missing data imputation techniques: A review. *JOIV: International Journal on Informatics Visualization*, 6(1-2): 278–283.

Stekhoven, D. J.; and Bühlmann, P. 2012. MissForest—nonparametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1): 112–118.

Sun, Y.; Li, J.; Xu, Y.; Zhang, T.; and Wang, X. 2023. Deep learning versus conventional methods for missing data imputation: A review and comparative study. *Expert Systems with Applications*, 227: 120201.

Tseng, H.-Y.; Jiang, L.; Liu, C.; Yang, M.-H.; and Yang, W. 2021. Regularizing generative adversarial networks under limited data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7921–7931.

Yin, D.; Kannan, R.; and Bartlett, P. 2019. Rademacher complexity for adversarially robust generalization. In *International conference on machine learning*, 7085–7094.

Yoon, J.; Jordon, J.; and Schaar, M. 2018. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, 5689–5698.